PGPre version 2 for Alpha Five

Overview

PGPre (Pee-Gee-Pree) is a general purpose utility application for Alpha Five that allows an Alpha Five (Alpha5) application developer to externally compile Alpha5 Xbasic scripts and/ or to maintain version control on these scripts using a version control system. All this can be done simply and effectively within any Alpha5 desktop database application from a single PGPre installation. For example:

I. External Compile

PGPre allows an Alpha5 developer to compose, modify and then syntax check Alpha5 Xbasic global scripts and functions from within any external text editor or development environment. Optionally the Alpha5 compiler output can be captured by ***PGPre*** and automatically re-displayed in the text editor's output window.

II. Version Control

Any Alpha5 Xbasic global scripts and functions can be recorded in a working folder and easily moved to and from between an Alpha5 application and its specified working folder. ***PGPre*** also allows scripts to be viewed, compared and synchronised between the application and its working folder. Whenever it is required, the working folder contents can be checked into the version control system.

PGPre provides an interface between Alpha5 and this external development environment or editor and makes it very easy and quick for the developer to develop and test Xbasic code externally to Alpha5. ***PGPre*** allows the developer to work with a preferred development application editor and use the development facilities there to which they are accustomed. Some of these facilities may be unavailable in the Alpha5 built-in editor - for example code folding, split screen editing, text drag-and-drop, code templates, project based work spaces, etc.

PGPre is flexible in the manner it can be used. For example it can be used for both external compiles and for version control or for just one or other of these features alone.

About Version 2

PGPre version 2 is a major update to the original release of ***PGPre***. It improves the overall usability of the product and adds numerous features to facilitate Alpha5 Xbasic script compile, import, export and compare. This can all now be accomplished via an improved workflow and re-designed menus and screens to make working in ***PGPre*** much easier.

New features added or improved include:

- Re-designed menus. The main menus are now able to be completely hidden too.
- Script imports, exports have been added and substantially upgraded from the original "Backsave" feature.
- Script compares overhauled and substantially improved.
- Settings menu added where more options and preferences can now be selected, altered and saved.
- Added support for external version control systems (VCS) for Xbasic scripts.
- More PGPre pre-compiler directives have been added. Some can be altered directly from the settings menu.

- Alpha5 compiler output can now be automatically captured and displayed in an external editor's output pane.
- Intelligent processing of password protected scripts so that the developer is not continually prompted by Alpha5 for passwords on opening protected scripts.
- And a lot more too.

About version 2.1

PGPre version 2.1 improves the overall reliability and adds new features. The compare, import, export and delete of scripts and script text files have been amalgamated into a single screen "Actions" panel display where formerly these were all separate. The ***PGPre*** menu items have as a consequence been simplified. The Actions display panel now shows both scripts and matching script text files in the same list and whether they match or differ.

For a summary of these and other changes refer to the release notes at the end of this document.

PGPre Requirements

PGPPre pre-requisites are as follows:

- Microsoft Windows (e.g. 2000, XP, Vista).
- Alpha5 Developer, version 8[1] or later.
- Any third party text editor or development environment – for the external compile features of ***PGPre***.
- Any third party version control system (VCS) – for the version control features of ***PGPre***. (There are many VCS product options here, some are free.)
- A third party file compare product – for the script and file comparison features of ***PGPre***. (There are many third party options here, some are free.)

A third party editor such as UltraEdit or UEStudio or Zeus for Windows is ideal. Notepad++ can be used too. There are many others. Practically any such editor can be used with ***PGPre*** so long as it has the ability to save code syntax in plain text format and run the ***PGPre*** Windows executable and include a parameter that supplies the full quoted name of the Alpha5 Xbasic code file.

A VCS such as the "Subversion" based TortoiseSVN (fee) or others such as QVCS can be used with ***PGPre***. There are many to choose from and which one you decide on will be determined largely by personal preference or corporate standards.

PGPre is pre-configured with interfaces for many third party products including a number of compare products: WinMerge (free), UltraCompare Professional, Beyond Compare 2 and Beyond Compare 3. Others can be provided as required. (All that is required is documentation of the product's command interface and a free trial version of the software so it can be tested – no re-installation of PGPre is then required).

[1] For maximum compatibility with PGPre, the latest release of Alpha5 version 8, 1980-3163 (released on 27 March 2008) is recommended or any Alpha5 version 9 release. Alpha5 version 8, 1965-3156 is also tested and confirmed compatible. Earlier versions may or may not be compatible with PGPre. PGPre checks your installed copy and will advise either an error or warning, as appropriate, if your version of Alpha5 is not, or may not be, compatible. In this event you can download and install the latest version 8 update from the Alpha Software web site, for example: http://downloads.alphasoftware.com/A5V8Download/PatchDownload.ASP

Installation

PGPre is supplied as a setup installation executable file for Microsoft Windows. When run the setup installer copies files to the following locations on your hard disk:

- ***PGPre*** pre-compiler Windows executable is copied to the Programs folder.
- ***PGPre*** pre-compiler Alpha5 add-in library is copied to the Alpha5 addins folder.
- ***PGPre*** user manual (this document) is copied to the application "Doc" folder and a Windows desktop shortcut created.
- A Windows desktop shortcut is created to run ***PGPre*** and Alpha5.

Initial Configuration

No configuration for any Version Control System is required for ***PGPre***. Configuration is required for file compare products and those that are already provided are as stated above under the "PGPre Requirements", the current status of others is available on request.

Before an Alpha5 compile can be run from an external editor, the editor must be manually configured. This can be done in the following manner:

- Configure the Tool or Run settings to be able to run ***PGPre*** via a command.
- Add the Tool or Run command you have created to the menu structure and/or toolbar.

For example for UltraEdit and UEStudio, setting up the command is done from the "Advanced/ Tool Configuration" menu item. Adding the command execution to the menu structure is done automatically. Adding the command execution to the toolbar is done by right clicking on any unused area of the toolbar and selecting "Customize Toolbar…". Once done you will end up with a "Run Alpha5 compile" icon on the toolbar much like the one illustrated in figure 6.

The Tool Configuration settings for UltraEdit/ UEStudio will be similar to the following and are accessible from the "Advanced/ Tool Configuration" menu:

Command line: **"C:\Program Files\PGSD\PGPreCompiler\PGPreCompiler.exe" "%f"**

Toolbar bitmap… **C:\Program Files\PGSD\PGPreCompiler\Alpha5.ico**

Standard and "Capture Output" Configurations

Two configurations are supported by ***PGPre***:for use with a third party editor:

The Standard configuration, and

The configuration for "Capture Compiler Output"

The Capture Compiler Output configuration can be used only with third party editors that support this feature. It allows the Alpha5 compiler output to be captured by PGPre and automatically re-displayed back in the editor's output pane. For this feature to work the ***PGPre*** capture output setting must be enabled, either on the Settings screen or via a compiler directive. (Refer to later sections to see how this can be done – "Settings" and "PGPre Precompiler Directives" sections).

The Capture Output pane for UltraEdit/ UE Studio can be viewed after selecting the "Output Window" item on the Window menu.

The following table shows how to set up UltraEdit/ UE Studio for the Standard and Capture configurations.

UltraEdit and UE Studio Configuration for use with PGPre

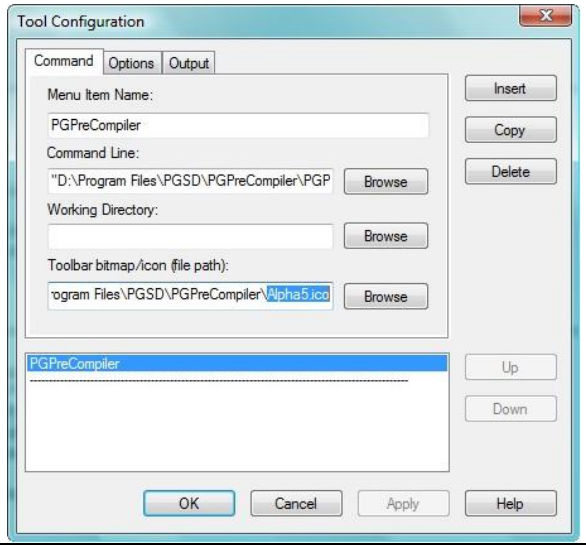
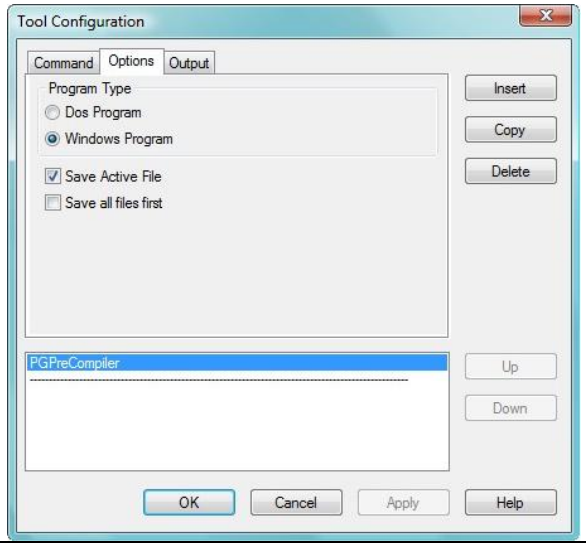
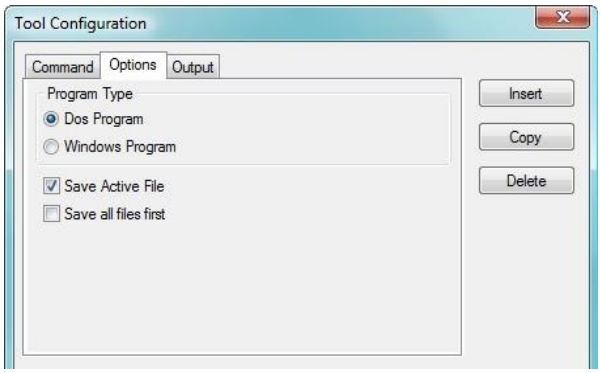
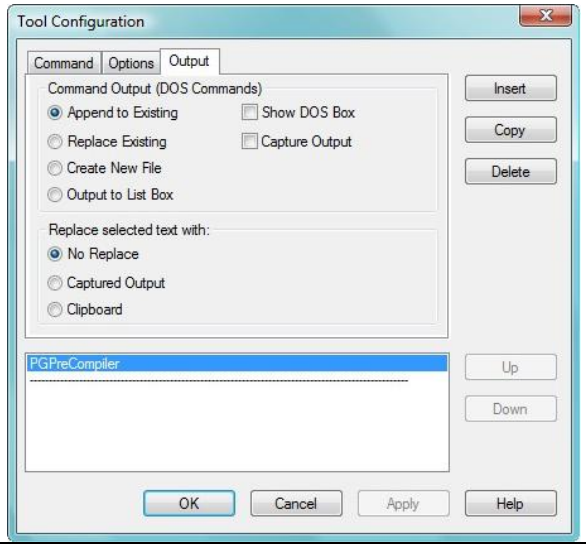
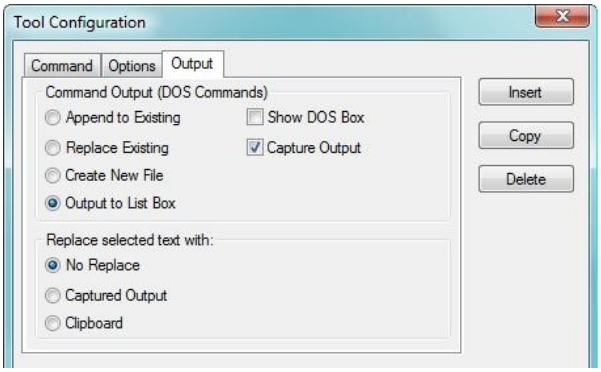
Standard Configuration	"Capture Compiler Output" Configuration
	<-- Same as for the standard configuration on left.
	
	

Figure 1: *PGPre* "Tool Configuration" Settings for UltraEdit/ UEStudio.

For Notepad++ setting up the command execution is done from the “Run/ Run… Save…”menu item. The syntax to apply will be the same or similar to the following, depending on where you installed ***PGPre***. An example of this syntax is as follows:

“C:\Program Files\PGSD\PGPreCompiler\PGPreCompiler.exe” “$(FULL_CURRENT_PATH)”

Copy this information into the dialog box beneath the heading "Type your program to run here". Both sets of double quotes are mandatory. Save this syntax with the name “Alpha5”. Modifying the Notepad++ Alpha5 setting can be done from the “Settings/ Shortcut Mapper/Run Commands” menu items. Note that there is no capability in Notepad++ to add a Run icon to the Notepad++ toolbar, as there is with UltraEdit/UEStudio. Notepad++ does not support “Capture Compiler Output”.

Configuration for Other 3rd Part Editors

If you are using another third party editor the initial configuration will be different. Use the above examples as a guide.

How to Start PGPre

It would be nice to be able to run ***PGPre*** from a toolbar icon within the Alpha5 Code Editor. Unfortunately that is not readily done so here are a number of alternatives.

1. Run ***PGPre*** and Alpha5 from the Windows desktop shortcut created by the supplied setup installer.
2. Run a compile from your third part editor using the menu item or toolbar icon you created in the Initial Configuration section above.
3. In the Alpha5 Code Editor go to the Interactive pane and type **“?PGPre()”** (with the question mark and without the quotes) and press the enter key.
4. Attach the following Xbasic code to a hidden hotspot on one of your applications forms, for example the “About this application” or “Settings” forms:

```
if "Developer" $ version("E") then
    PGPre()
end if
```

A fifth and probably the most useful way to start ***PGPre*** is to go to the Code tab of the Alpha5 Control Panel (developer’s version only), right click in any of the white background display and from the Addins menu item click on and run the PGPre Compiler item (see figure X). This will run ***PGPre().***

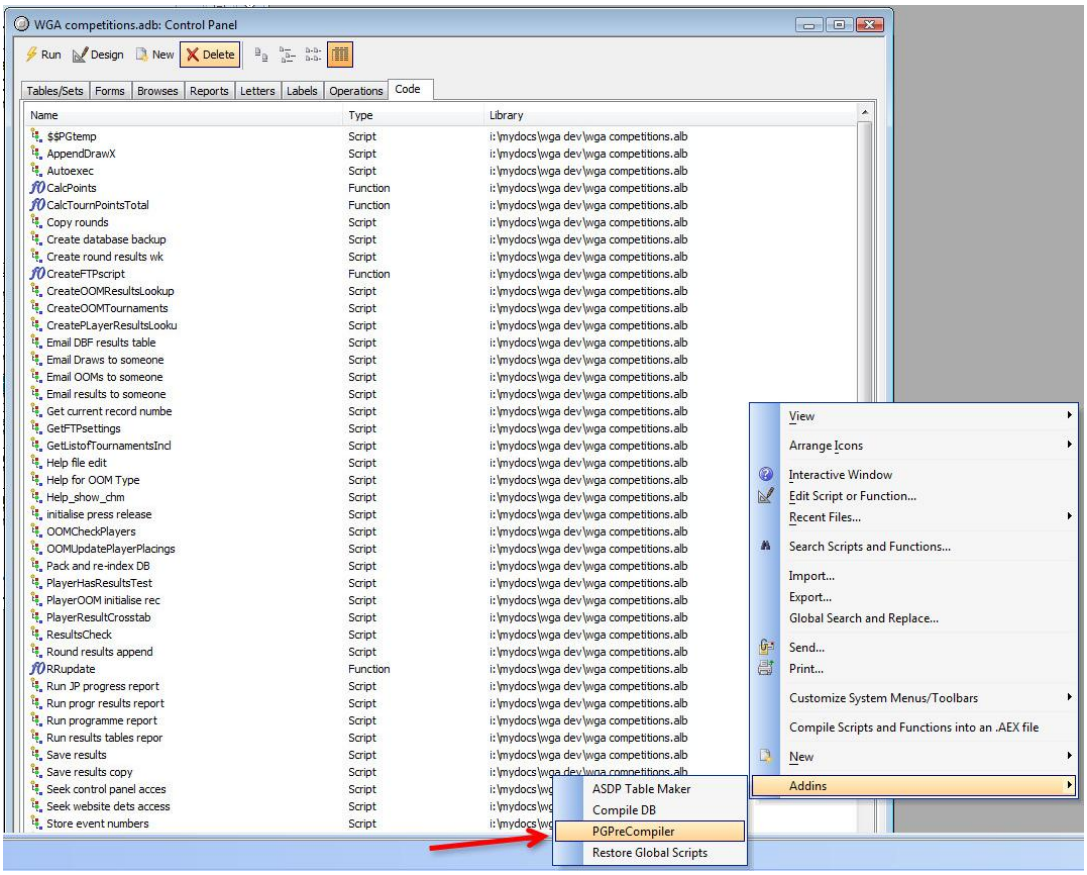


Figure X: Run PGPre from the Alpha5 Control Panel Code Tab

If the "PGPre Compiler" option does not appear in the list shown in figure X, close down your copy of Alpha5 and copy the file PGPRECOMPILER.AEX as per the following examples. Then restart Alpha5 again.

Copy PGPRECOMPILER.AEX from e.g. C:\Program Files\PGSD\PGPreCompiler\Addin

Copy PGPRECOMPILER.AEX to e.g. C:\Program Files\A5V8\addins_installed

Any of these five options will start PGPre and present its main menu – see Figure 3.

PGPre Pre-Compiler Directives

The behaviour of ***PGPre*** can be modified by using Directives. These Directives may be included in your Alpha5 Xbasic script at the start. The rules for creating the Directives are as follows:

- Directives must be within the first comment block (no Xbasic code or blank lines must precede them or the Directives will be ignored).
- There must be only one Directive per line.
- The line must contain '# (a single quote followed by a hash symbol) in columns 1 & 2.
- The Directive name must then appear and be followed by an equal sign =
- The value or text you wish to assign to the Directive must appear after the equal sign.
- If the value is text, you should enclose the value within double quotes.
- If there is no value or text following the equal sign, or a Directive is omitted, default values will apply – see below.

The Directives are described as follows.

Directive Name	Default Value	Can be Altered in *PGPre* Settings Menu	Description
A5CaptureOutput	"no"	yes	Capture Alpha5 compiler output and present in the external editor's output pane (requires a compatible editor and the correct configuration of it – see Initial Configuration section).
A5Close	"no"	yes	Close Alpha5 after completion of each compile. If "yes" this will take precedence over the Timeout Directive setting.
A5DB	If PGPre is running, the currently open Alpha 5 .ADB file. If PGPre is not running the latest Alpha5 .ADB file previously opened.	n/a	The full name (path, file name and extension .ADB) of the A5 application the Xbasic code file is required to be opened in. Must be enclosed in double quotes, e.g. "I:\MyDocs\Alpha5v8\TestIt1\TestIt1.adb"
IgnorePGPreTimeStamp	"yes"	yes	Note: this directive can only be set from the ***PGPre*** Settings screen. Its purpose is to allow the line containing the ***PGPre*** time stamp to be omitted from a script compare so that a script and its text file that are equal but for the ***PGPre*** time stamp will still show in a comparison summary as being equal.
A5Minimize	User specified default	yes	Minimize Alpha5 on completion of each compile and let ***PGPre*** automatically detect and run further compiles as each is initiated.
A5ScriptName	File name of Xbasic code file	n/a	The name that the Xbasic code is to be saved as in the A5 .ADB application. If more than the Alpha5 limit of 24 characters is present, they will be truncated after 24th. Must be enclosed in double quotes, e.g. "My test script".
TimeOut	0	n/a	The maximum time in seconds that A5 will remain open to receive the next compile. Zero will leave Alpha5 open indefinitely or until the user terminates via the Close buttons. Whenever a timeout occurs, the option is provided to either close Alpha5 or continue and invoke another timeout period.
TimeStamp	"yes"	yes	***PGPre*** will place a date and time marker in the text of any script when compiled, imported or exported.
A5Translate	"yes"	n/a	Some third part text editors (Notepad++ included) will not provide syntax highlighting

			and code folding features for Xbasic language elements that include space characters. Examples are “end if”, “end while”, “end function”, etc. With this Directive enabled, language elements may be coded without the spaces (endif, endwhile, endfunction etc.) and ***PGPre*** will automatically convert them to the Alpha5 correct syntax.
A5VersionNumber	Latest installed Developer version of A5	n/a	***PGPre*** will run a compile using the latest installed version of Alpha5 Developer. If there is more than one version installed, another version may be specified here. For example, if versions 7, 8 and 9 of Alpha5 Developer are installed, versions 7 may be specified.
A5WarnIfExists	“yes”	yes	When a compile is run if the Alpha5 script name already exists, a warning message will be issued before it can be replaced. If you accept the default name for your script ($$PGtemp), you will not be prompted.

Figure 2: PGPre Directives

PGPre Script Template Example

Shown below is a sample ***PGPre*** script template for developing Alpha5 Xbasic that can be used with any third party product that supports templates.

```
‘#-------------------------------------------------------------------------
‘# PGPre Directives
‘#-------------------------------------------------------------------------
‘#A5CaptureOutput =
‘#A5Close =
‘#A5DB =
‘#A5Minimise =
‘#A5ScriptName =
‘#TimeOut =
‘#TimeStamp =
‘#A5Translate =
‘#A5VersionNumber =
‘#A5WarnIfExists =
‘=========================================================================
‘ Script name:  <insert your script name here – documentation only>
‘
‘ Date Written: <insert date and time variable here>
‘
‘ Written by:   <your name and company goes here>
‘
‘ Copyright:    <your name and company goes here>
‘               <other goes here>
‘
‘ Purpose:      <your documentation goes here>
‘
‘=========================================================================
‘ Your own Alpha5 Xbasic code will start here.
```

Using PGPre

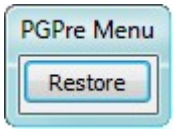
I. External Compile

To run a compile of your A5 Xbasic script from within your third party editor all that is required to initiate a compile is to click on the 'Run Alpha5' Compile menu item or toolbar icon you created in the "Initial Configuration" section above. This starts up, runs Alpha5 and a compile of your current Alpha5 Xbasic script using any ***PGPre*** Directives specified in your script.

You can leave ***PGPre*** running in the background (minimized recommended) if you intend to run further compiles and do not want the overhead of continually re-starting and stopping Alpha5 for each compile. Just use the **A5Close = "no"** and **A5Minimize = "yes"** Directives to do this. One of the following three floating dialog boxes will be displayed while ***PGPre*** is awaiting the next compile:

Main – PGPre Menu 1

Relocated – PGPre Menu 2



Minimized – PGPre Menu 3

Figure 3: *PGPre* Menu Alternative Layouts 1 (top), 2 (middle), 3 (bottom).

You can continue to work normally in Alpha5 doing other things while ***PGPre*** is running in any of these modes with any of the illustrated three dialog boxes displayed – ***PGPre*** uses no resources in doing this.

The displayed dialog box option buttons are as follows:

Button Name	Description
Relocate PGPre Menu	Only visible on ***PGPre*** Menu 1, this button relocates the ***PGPre*** Menu to the top left of the screen and changes its appearance to the reduced Menu 2 display layout.
Minimize	Only visible on ***PGPre*** Menu 2, this button reduces the menu to the Minimized format of the Menu 3 display layout.
Restore	Only visible on ***PGPre*** Menu 3, this button reinstates the Menu 1 display.
Hide PGPre Menu	Hides the ***PGPre*** menus (1, 2 and 3). The menu will be shown again when another compile is initiated. Alternatively go to the Alpha5 Code Editor Interactive window and type "**?PGPre()**" without the quotes and press the enter key.
Actions	Shown only on menus 1 and 2, this menu item gives access to the combined features of PGPre to compare, import, export and delete scripts and script text files.
Settings	Provides for a number of settings and defaults to be altered and saved. These settings include some ***PGPre*** directives, a default working folder, default script text file extension (.axb or .txt) and setting up a Compare product for use with PGPre.
Help	Displays the ***PGPre*** documentation PDF file (this document).
About	Displays the "About" dialog box.
Close PGPre	Terminates ***PGPre*** and leaves the current Alpha5 database open.
Close current DB	Terminates ***PGPre*** and closes the current Alpha5 database without closing Alpha5.
Close A5	Terminates ***PGPre***, closes the current open Alpha5 database and closes Alpha5.

You can elect at any time to close ***PGPre*** or close Alpha5 by clicking on the appropriate close button. With ***PGPre*** closed it will need to be re-opened to initiate another compile. This can be achieved by closing Alpha5 and initiating a compile from your third party editor or developer environment. ***PGPre*** will prompt you whenever this is necessary.

II. Version Control

Another main use of ***PGPre*** is to maintain version control of Xbasic global scripts and UDF script global functions. There are numerous ways to do this but they essentially involve maintaining an application working folder for scripts (one folder per Alpha5 application) and moving the scripts between an Alpha5 application and its working folder using the Import, Export and Compare features of ***PGPre***.

The application working folder for scripts is also defined as the version control system's (VCS's) working folder. From there scripts can be checked in and out as required, for

example on each application version change. In this way version control can be maintained on all scripts to show:

- When changes were made to each script
- Exactly what the changes are between any two versions
- The reason for any changes.

Scripts may also be reverted to any prior version and the application re-instated to this point. Many VCSs allow you to specify a favorite compare utility application so you can view all changes within the tool you are accustomed to working with.

Here are a few of scenarios showing how a VCS can be used with ***PGPre*** and Alpha5. These scenarios are based on the assumptions that:

- I have an existing version 1 application that I have just completed work on of a major upgrade of it to version 2.
- The version 1 scripts are recorded in the VCS repository and not yet updated to version 2.
- My version 2 scripts are of three main types – those that have been modified from version 1; new scripts added for version 2; scripts deleted and no longer present in version 2.

Scenario 1 – No scripts permanently reside in Alpha5, All are held in the VCS

Normally I have no scripts residing in my Alpha5 application. They normally reside in the VCS repository and my application contains only the application's compiled AEX library.

As I require the version 1 scripts to work on in version 2, I will check out those scripts that I need from the VCS repository into the working folder and work on and test them from there in either the Alpha5 Code Editor or my third party editor. With ***PGPre*** I can move scripts between the working folder and Alpha5 as required, in either direction, and I can also see at a glance just what the overall differences are between the two. If I need to see the detailed differences I can run a Compare in ***PGPre*** and view the detailed differences between any script in Alpha5 and its equivalent copy in the working folder.

Note that I can continue to run my old version 1 and version 2 scripts side by side even though the version 1 scripts do not reside in my Alpha5 application – recall that they are only in the VCS repository and nowhere else. Alpha5 will automatically run the version 2 scripts and only run version 1 scripts from the version 1 AEX compiled library when there is no version 2 script present. I will not compile another AEX library again until I need to produce one for version 2.

When I have completed work on version 2 of my application and am ready to compile it to a new version 2 AEX library and to shift the latest version 2 scripts to the VCS repository, I need to move the unchanged version 1 scripts that are still required into version 2, from the VCS to the working folder making sure I don't inadvertently select and overwrite one – VCS does this automatically for me. I then need to move these into my Alpha5 application – which is made very easy by ***PGPre*** – and recompile a version 2 AEX library for my application. Having done that I can just delete all my Alpha5 applications scripts! I use the 'Delete scripts' feature of ***PGPre*** to do this because it will tell me anytime I am deleting a script that does not already exist in or does not match the copy in my working folder. It will also tell me before I delete the scripts exactly which ones differ or are missing.

Finally I check my version 2 scripts back into my VCS from the working folder and then delete them all from the working folder.

Scenario 2 – I use my Working Folder as My personal Script Repository

This scenario is arguably best suited for a single-developer. It's somewhat easier to maintain than scenario 1 because there is not the complication of multiple developers wanting to share and use the same scripts at the same time. Think of it this way. All an Alpha5 application's scripts are permanently checked out to the working folder from the VCS repository. The VCS contains the version 1 scripts, as does the working folder. When I start work on version 2 I first verify in the VCS that the working folder and VCS repository are the same. When they are (and they should always be – if not just use the VCS to determine why and synchronise changes) I can begin work on version 2.

I can then work on updating existing and creating new scripts in either Alpha5 or my third party editor. I can compare and move scripts as required between the working folder and Alpha5 using ***PGPre***, modify and test them as required.

When I am completed I can use ***PGPre*** to quickly compare my scripts in Alpha5 with those in the working folder and make any adjustments by importing or exporting those that are required by using ***PGPre***. Then all I need do is to create the version 2 compiled AEX library and delete all the scripts from Alpha5 using ***PGPre***'s 'Delete scripts' feature. Recall that I can delete scripts with confidence because ***PGPre*** will tell me twice, both before I delete scripts and when I actually try to delete them whether or not the copies in Alpha5 that I am deleting are the same as those in the working folder.

Finally I can get my VCS to copy all the scripts from the working folder to the VCS repository and I'm all finished. My working folder and VCS repository are now synchronised on version 2. If I want to see changes between version 1 and version 2 scripts I can see everything I want, including all changes between versions 1 and 2, in the VCS system.

Scenario 3 – I Complete work on a New Alpha5 Application

This is the simplest scenario of all. When I have completed my application (let's say it's version 1) I can create a working folder (if I have not already done so) and compare and/or move scripts there from Alpha5 using ***PGPre***. I compile my application to an AEX library file and then delete all the scripts from Alpha5. Finally all I need do is copy all the working folder script text files into my VCS and I am finished. My application is now under VCS control where I can see any future version and the detailed contents and status of any script as it was at any previous point in time.

Actions Menu Item

The actions menu item combines many of the main features of PGPre into a single screen display panel.

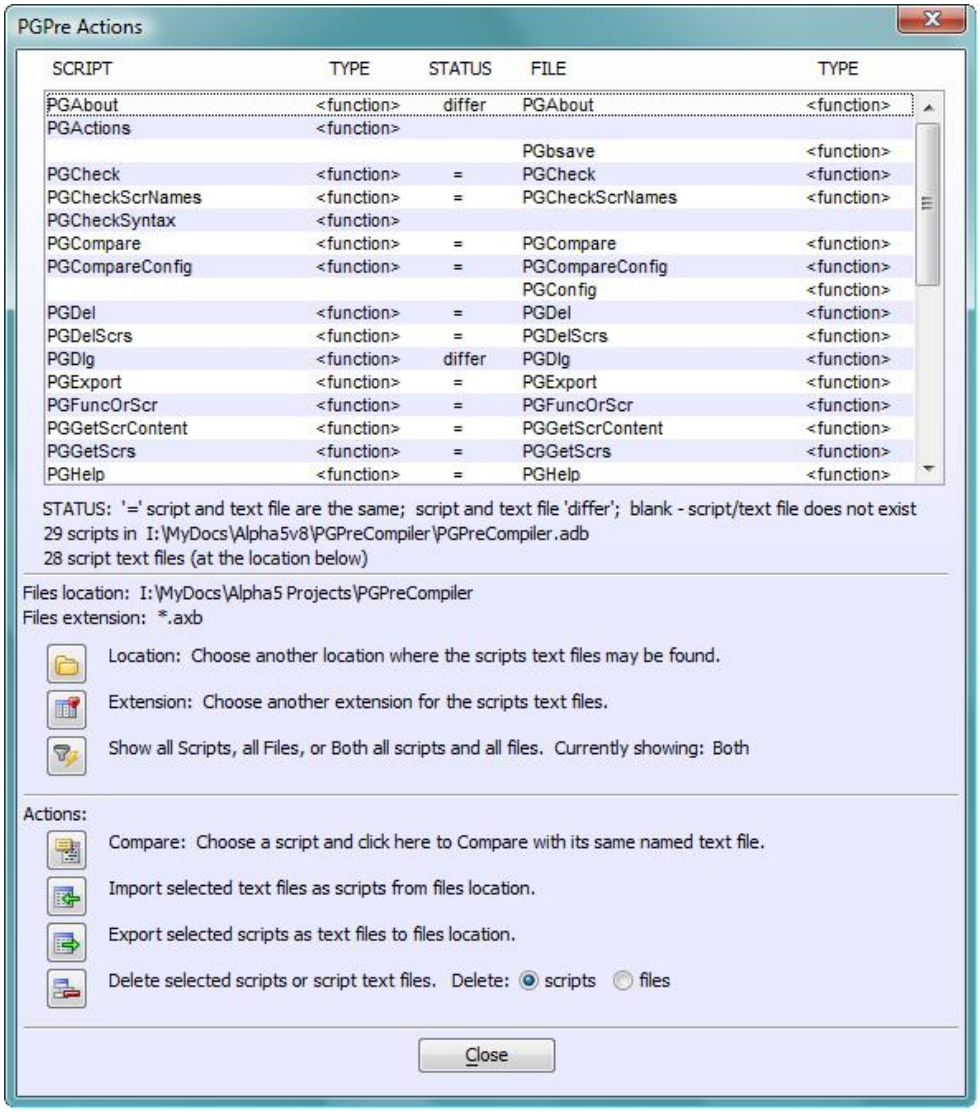


Figure 4: PGPre Actions

The list box in the top half of the display (figure 4) shows:

On the left hand side	Global scripts and user defined functions in the currently open Alpha5 application .ADB file.
On the right hand side	All text files in the current selected working folder with the chosen extension of either .axb (Alpha5 Xbasic) or .txt.

The display can be restricted to just all scipts/ functions or all text files or both all scripts and all text files using the Show button. This will cycle the display through the three display options as the button is clicked.

The screen buttons have the following functions.

Location: Select another temporary folder location from which to select script text files for display. The default location for the scripts is the application's current working folder (specified in the ***PGPre*** Settings screen).

Extension: Choose the file extension of files for display, either .axb (Alpha5 XBasic) or .txt. The default is the extension selected and saved on the ***PGPre*** Settings screen.

Cycles the screen display between the three name display formats – show all scripts, show all files or show both all scripts and all files.

Compare: If you want to see the exact differences between a text file and its matching script file, click on the corresponding row in the list box and click on this button. The files and any differences will be shown in the Compare product of choice selected sand saved from the Settings screen. Refer to the following section for further information.

Import: Select any or all text files that you want to import as scripts into your current A5 application, then click on this button to do the import. Individual and/or ranges of selections can be made using the standard Windows multi-select keyboard commands – Ctrl-click for individual selections, Shift-click for ranges, click-hold-drag-release for ranges.

Export: Select any or all scripts that you want to export to the current working folder selection, then click on this button to do the export. Multiple selections can be made as described under Import.

Delete: Selected scripts or files may be deleted by checking the scripts or files option and then the scripts or files to be deleted. Click on this button to then do the deletion. Multiple selections can be made as described under Import.

Import Limitations

Alpha5 script names are limited to 24 characters, Windows file names are not. In allocating a name to an imported script ***PGPre*** will truncate a file name to 24 characters if the file name exceeds this limit. If the selection of files includes names that exceed 24 characters, ***PGPre*** will truncate them to the first 24 and if this results in two or more with the same name a warning message will result. The options are to continue but ignore such duplicates (the first will be accepted) or cancel.

Export Limitations

Scripts are exported as script text files to the folder and with the file name extension selected as described above. Alpha5 script names may include characters that are not permitted in Windows file names. There are nine such characters as follows:

Invalid Windows file name characters: **\ / : * ? “ < > |**

PGPre will check for script name selections that contain any of these characters. The Export will be curtailed if such selections are found by ***PGPre***.

Translate Converted from	Translate Converted To
DefineClass	Define Class
EndClass	End Class
EndIf	End If
EndSelect	End Select
EndFunction	End Function

EndType	End Type
EndWhile	End While
EndWith	End With

The Translate feature is the same as done by the ***PGPre*** Directive 'A5Translate' (see figure 2) when a script is saved. Translate is not case sensitive and so any variation of capitalization will be converted to the syntax shown in the right column.

Compare Script and Text File

The ***PGPre*** Actions "**Compare**" button (see figure 4) allows any script in the current Alpha5 database to be selected and then compared in detail with its associated script text file. Alternate temporary selections for the folder and extension may be made using the location and extension buttons provided.

Script Text File Status

The script text file status is a summary comparison of a script and its same named script text file. The comparison is not case sensitive (so "Abc" is the same as "abc") and all white space is ignored. White space includes blank lines, spaces, tabs and other noise characters not normally displayed by the editor you are using. The comparison optionally may also exclude any added ***PGPre*** date and time stamp that may otherwise make the comparison non-identical – refer PGPre Settings display panel in figure 7.

To compare a script with its associated text file, after making a selection from the list box in Figure 4, click on the Compare button . A dialog box will allow for confirmation of your choice as follows.

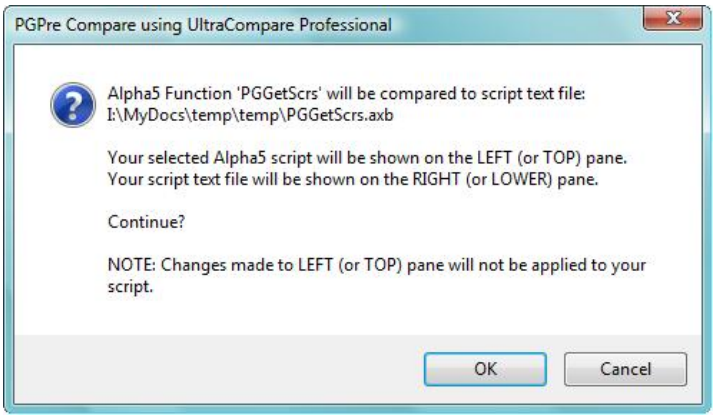

PGPre Compare using UltraCompare Professional
Alpha5 Function 'PGGetScrs' will be compared to script text file:
I:\MyDocs\temp\temp\PGGetScrs.axb
Your selected Alpha5 script will be shown on the LEFT (or TOP) pane.
Your script text file will be shown on the RIGHT (or LOWER) pane.
Continue?
NOTE: Changes made to LEFT (or TOP) pane will not be applied to your script.
OK
Cancel

Figure 5: Compare confirmation dialog box

At this point you may choose to cancel or proceed to the comparison which will use the Compare product selected in the Settings screen. Here is an example of the comparison.

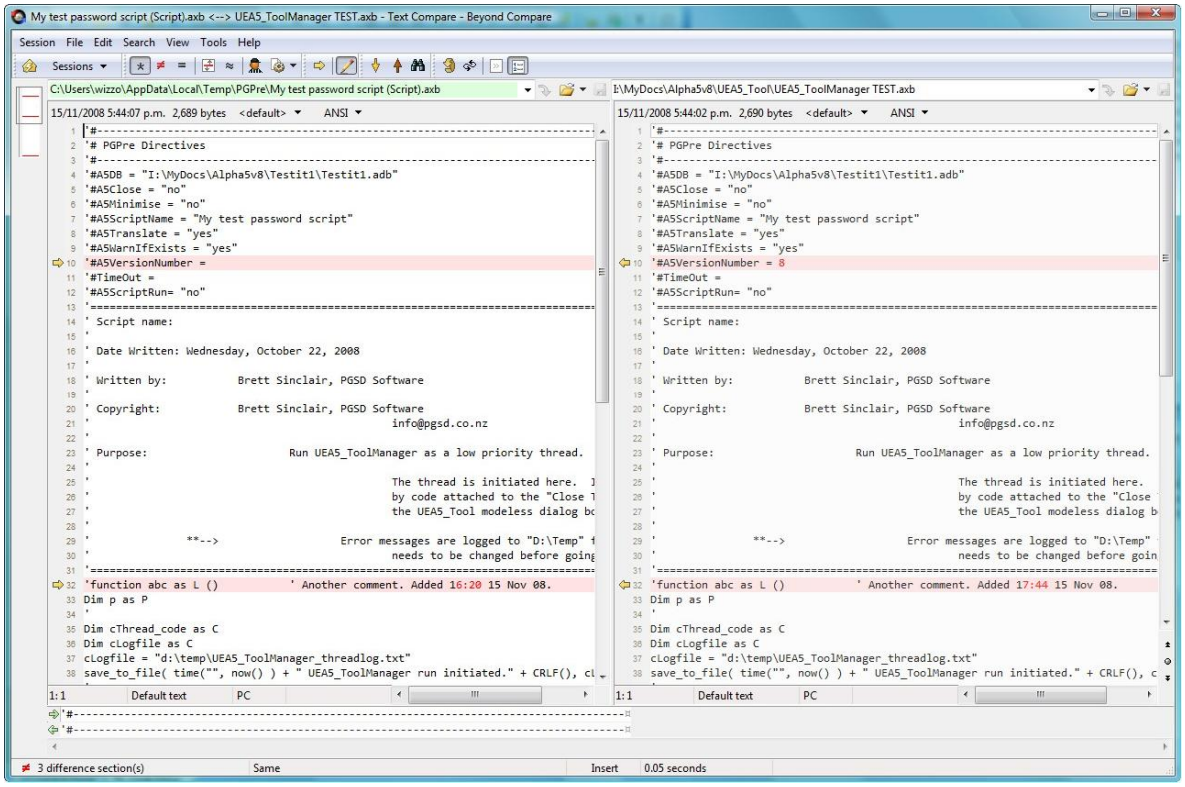

My test password script (Script).axb <--> UEA5_ToolManager TEST.axb - Text Compare - Beyond Compare
Session File Edit Search View Tools Help
Sessions
C:\Users\wizzo\AppData\Local\Temp\PGPre\My test password script (Script).axb
I:\MyDocs\Alpha5v8\UEA5_Tool\UEA5_ToolManager TEST.axb
15/11/2008 5:44:07 p.m. 2,689 bytes <default> ANSI
15/11/2008 5:44:02 p.m. 2,690 bytes <default> ANSI
'# PGPre Directives
'#A5DB = "I:\MyDocs\Alpha5v8\Testit1\Testit1.adb"
'#A5Close = "no"
'#A5Minimise = "no"
'#A5ScriptName = "My test password script"
'#A5Translate = "yes"
'#A5WarnIfExists = "yes"
'#A5VersionNumber =
'#A5VersionNumber = 8
'#TimeOut =
'#A5ScriptRun= "no"
' Script name:
' Date Written: Wednesday, October 22, 2008
' Written by: Brett Sinclair, PGSD Software
' Copyright: Brett Sinclair, PGSD Software
info@pgsd.co.nz
' Purpose: Run UEA5_ToolManager as a low priority thread.
'function abc as L () ' Another comment. Added 16:20 15 Nov 08.
'function abc as L () ' Another comment. Added 17:44 15 Nov 08.
Dim p as P
Dim cThread_code as C
Dim cLogfile as C
cLogfile = "d:\temp\UEA5_ToolManager_threadlog.txt"
save_to_file(time("", now()) + " UEA5_ToolManager run initiated." + CRLF(),
1: 1 Default text PC
3 difference section(s) Same Insert 0.05 seconds

Figure 6: Compare results – an example

The step compare results are shown in your Compare Application of choice (see the following Configure section) in the following order:

- **Left (or Top)** Your Alpha5 selected script from your currently open Alpha5 database.
- **Right (or Lower)** Your selected script text file.

Note: Changes may be made to and/or merged with the right hand script text file but changes made to the left (or lower) Alpha5 script will **NOT** be applied to your Alpha5 selected script. A later version of ***PGPre*** may allow this.

Settings

There are a number of settings and features that may be altered and saved from the ***PGPre*** Settings screen.

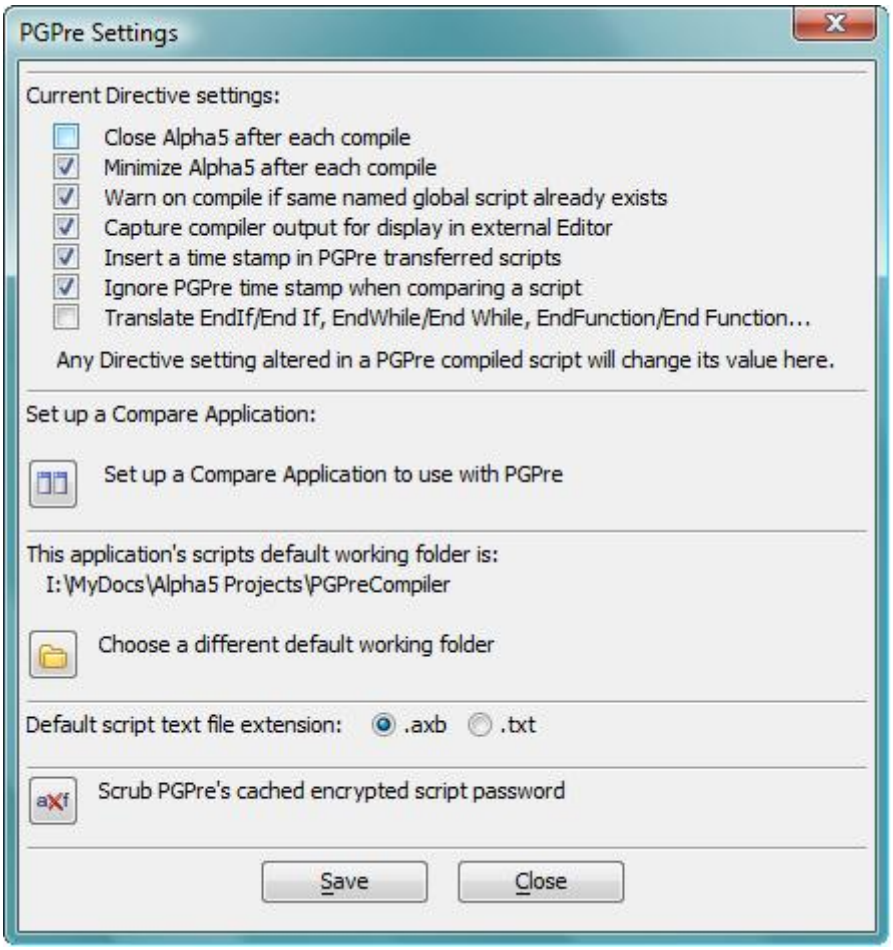

PGPre Settings
Current Directive settings:
Close Alpha5 after each compile
Minimize Alpha5 after each compile
Warn on compile if same named global script already exists
Capture compiler output for display in external Editor
Insert a time stamp in PGPre transferred scripts
Ignore PGPre time stamp when comparing a script
Translate EndIf/End If, EndWhile/End While, EndFunction/End Function...
Any Directive setting altered in a PGPre compiled script will change its value here.
Set up a Compare Application:
Set up a Compare Application to use with PGPre
This application's scripts default working folder is:
I:\MyDocs\Alpha5 Projects\PGPreCompiler
Choose a different default working folder
Default script text file extension: .axb .txt
Scrub PGPre's cached encrypted script password
Save
Close

Figure 7: Settings screen

I. Directives

Selective directives can be set on the Settings screen, those as shown. For an explanation of these directives, refer to the Figure 2.

The Translate feature is the same as done by the ***PGPre*** Directive 'A5Translate' (see figure 2) when a script is saved. Translate is not case sensitive and so any variation of capitalization will be converted to the syntax shown in the right column.

Translate Converted from	Translate Converted To
DefineClass	Define Class
EndClass	End Class
EndIf	End If
EndSelect	End Select
EndFunction	End Function
EndType	End Type
EndWhile	End While
EndWith	End With

II. Set up a Compare Application

The ‘Set up a Compare Application’ button will display a list box of all PGPre pre-configured Compare products found on your computer. Here is a sample.

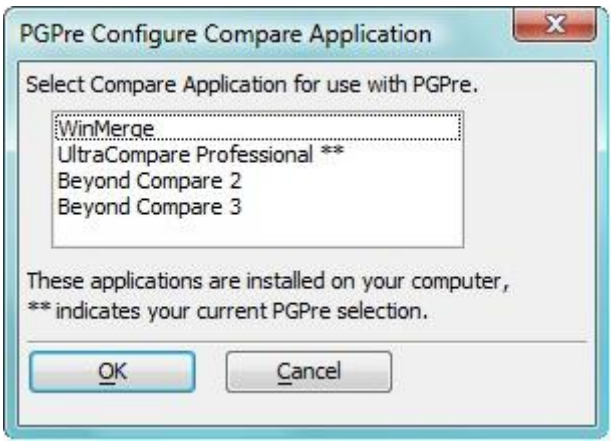


Figure 8: Select a Compare Application for Use with PGPre

PGPre will detect and list Compare/Merge Applications installed on your computer, presently limited to those displayed above. The current Compare/Merge Application already configured for use with ***PGPre*** (if any) is show with two asterisks as your current ***PGPre*** selection.

To select a different Compare Application for use with ***PGPre***, click on the item in the list shown and click on the ‘OK’ button.

Web site links from where these Compare applications can be downloaded and installed are listed at the end of this document. WinMerge is freeware, the remainder are pay-for products all with a 30 day free trial period.

III. Default Working Folder and Extension

A default working folder and extension for Xbasic script text files can be set here. These settings are used by the ***PGPre*** import, export, compare and delete features for the initial screen displays.

IV. Scrub Cached Password

PGPre maintains an application based password for use with password protected scripts. This password cache can be scrubbed clean by clicking on this button. Refer to the following section entitled “Password Protected Scripts” for further information.

Compiling and Saving Alpha5 Scripts

When you run a compile ***PGPre*** will identify and then save your Xbasic code in Alpha5 as either a **script** or **function**. If all the code is contained within a function block (appears between the “function” and “end function” statements) the code will be saved as a function. If any code (excluding comments) appears outside a function block, the code will be saved as a script. This also applies if the script contains more than one function block. Recall that Alpha5 allows you to have multiple functions in a single script. However, only the function that shares the script name will be externally available.

An example follows.

```
‘#-----------------------------------------------------------------------
‘# PGPre Directives
```

```
'#-------------------------------------------------------------------------
'#A5DB = "I:\MyDocs\Alpha5v8\TestIt1\TestIt1.adb"
'#A5Close = "no"
'#A5Minimise = "yes"
'#A5ScriptName = "AddOne2It"
'#TimeOut = 1800
'#A5Translate = "yes"
'#A5WarnIfExists = "yes"
'#A5VersionNumber = 8
'=========================================================================
' Script name:    AddOne2It() UDF
'
' Date Written:   31 October 2008
'
' Written by:     Brett Sinclair
'
' Purpose:        Adds 1 to the supplied number.
'
'=========================================================================
Function AddOne2It as N (nI as N)
      AddOne2It = nI + 1
End Function
'
Dim cJ as N
'=========================================================================
```

When it is saved, this Xbasic code will be saved by ***PGPre*** as an Alpha5 script and not as a function because the second-to-last line "**`Dim cJ as N`**" is code outside the function block. If this line was removed or commented out then the script would be saved as a function.

Password Protected Scripts

Password protection is afforded to scripts and functions stored in Alpha5 when the first line of the script contains syntax exactly of the following form:

```
'password mypassword
```

This format consists of:

- a single quote
- followed by the word "password", without the quotes and with no spaces preceding,
- followed by a single space and your password

Script passwords are not case sensitive and may not contain spaces. For further information refer to the section in the Alpha5 Help documentation entitled "Password Protecting Your Scripts".

The first time ***PGPre*** encounters a password protected script that is recorded in Alpha5 it will prompt you for the password. To prevent further repeated unnecessary and untimely requests, PGPre will not prompt you again – it will attempt to re-use that same password. Only if you are using different passwords in any single Alpha5 application will you be prompted for a password again.

PGPre records the latest script password in the application's library .ALB file. The password is securely encrypted and its original plain text will not be available to anyone else. To be doubly sure, there is a "scrub cached script password" entry on the PGPre settings menu so that the encrypted password can be deleted before the application is deployed. If

you forget to do this it doesn't matter, the information is securely encrypted anyway. But you can always re-deploy later another update of the application with the password scrubbed.

As a security measure, the global scripts and function of an Alpha5 application should not be deployed with an application. Instead deploy the scripts compiled to an .AEX library file and the scripts then deleted (or better still exported to their working folder and then deleted, all by using PGPre). For more information on compiling scripts to an .AEX library file refer to the Alpha5 documentation under the heading "Compiling Scripts and Functions".

PGPre Interactive Code Checker

PGPre mimics the feature of Alpha5 that allows Xbasic code snippets to be composed and checked in a scratchpad area separate from where you are working on your main code. When code is checked the following dialog box is displayed, for example:

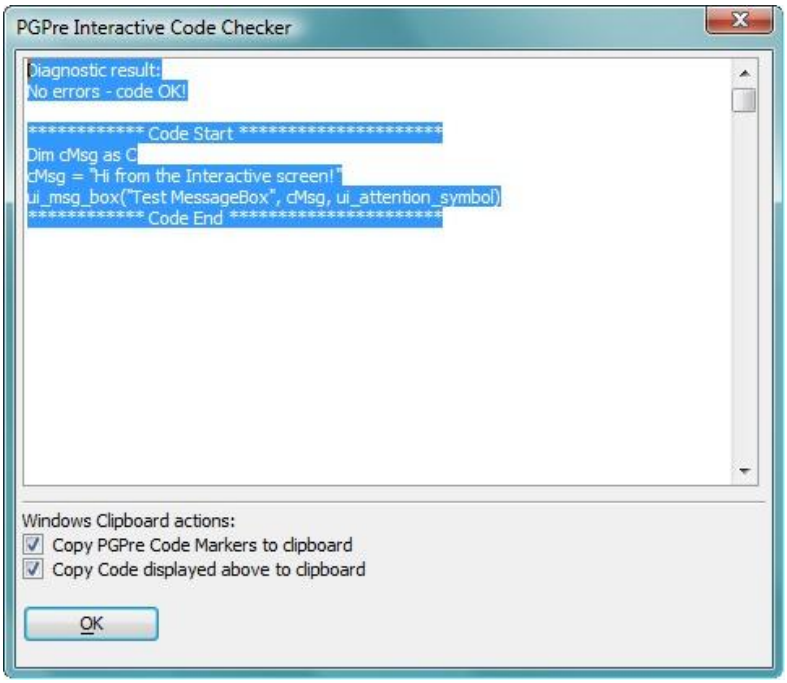

PGPre Interactive Code Checker
Diagnostic result:
No errors - code OK!
********** Code Start **********
Dim cMsg as C
cMsg = "Hi from the Interactive screen!"
ui_msg_box("Test MessageBox", cMsg, ui_attention_symbol)
********** Code End **********
Windows Clipboard actions:
Copy PGPre Code Markers to clipboard
Copy Code displayed above to clipboard
OK

Figure 9: The PGPre Interactive Code Checker

Your code is checked by Alpha5 and displayed along with any diagnostic error messages.

To make use of the PGPre Code Checker, follow these steps:

- In your third party text editor/development environment create and save a new file named "Interactive".
- Record in your "Interactive" file any code snippets you with to check and/or run.
- Optionally, place a start marker at the beginning and an end marker at the end of the code or lines of code you want to check.
- Click on the usual "PGPre Run Alpha5 Compile" toolbar or menu item.

The start and end markers are **<<%PGPre%** and **%PGPre%** respectively. Their use is optional and if omitted the entire code in your Interactive file is checked. You do not need to remember these markers because you can copy them to the Windows clipboard, with or without your code, by selecting the appropriate check box items on the PGPre Interactive Code Checker dialog box (see figure 10).

The interactive code checker will be run using PGPre Directive defaults. However you may specify in your Interactive file Directives that will override the defaults. The specified Directives do not need to be included between the begin and end markers. In normal circumstances it should be unnecessary to specify Directives in your Interactive file, but you can do so if you wish.

PGPre Usage Guidelines and Tips

Here are a few tips that should help you get the best out of your usage of PGPre. Some of these are obvious but many are not.

(a) Keep Each Alpha5 Application's Scripts in their own Location

To avoid confusion it's a good idea to keep each application's script text files in their own folder. Mixing scripts from more than one application in a single folder is bound to confuse.

(b) How to avoid Script Text File Naming Conflicts

Alpha5 restricts script names to 24 characters, Windows does not. To ensure you can import and export scripts with the minimum of fuss just make sure all your scripts are named uniquely and are no longer than 24 characters.

If you import a text file and its name is longer than 24 characters, ***PGPre*** will truncate the name to 24 characters and warn you if the name is already assigned to an existing script. However complications can arise, especially if you have two or more script text files with long names that are the same in the first 24 characters. ***PGPre*** will warn you when this happens but confusion will arise as to what text file the truncated name refers to. It's better to avoid this situation in the first place.

Alpha5 allows a script and a function to have the same name. In this situation when it comes time to export both to a common folder a naming conflict will arise. ***PGPre*** will warn you of this situation but it's more efficient if you can avoid the conflict in the first place.

(c) Avoid Using Special Characters in script names

There are nine special characters that Windows will not allow in a file name:

\ / : * ? " < > |

Avoid using them in Alpha5 script names. ***PGPre*** will warn you when it processes scripts with names containing these characters and give you the option of cancelling or skipping the script concerned. However it is much more convenient to avoid these characters completely.

(d) How to ensure PGPre Can Match up a Script and its Text File Correctly

PGPre pairs scripts and text files by name. For example script "My test script" and file "My test script.axb" located in the Alpha5 application's working folder will be paired by ***PGPre***. If you follow the previous suggestions you will be able to ensure your scripts and their text files are paired off correctly by ***PGPre***.

(e) How to Compare a script with a different text file

Because ***PGPre*** pairs off scripts with its same named text file, it is normally not possible to compare a script with a text file having a different name. However here is a tip that can allow you to accomplish this while you remain working in Alpha5 and ***PGPre***.

Select the 'Compare script' ***PGPre*** menu option. Any script that is listed on the ***PGPre*** Compare Scripts' screen (Figure 6) that has no matching file will show there with a narration 'no file'. Continue as follows:

- Click on such a file and click the Compare button.
- ***PGPre*** will tell you that the script's matching text file does not exist.
- There is a "Continue anyway?" option – just click the "Yes" button.
- ***PGPre*** will then start your file compare application with an empty script text file displayed on the right (or lower pane if you have it configured to display top and bottom instead of left and right.)
- Now all you need to do is browse to and select the text file you want to compare your script to and … that's it! Done.

Development Environment Example

The below example demonstrates the appearance and use of ***PGPre*** from within a third part text editor or development environment.

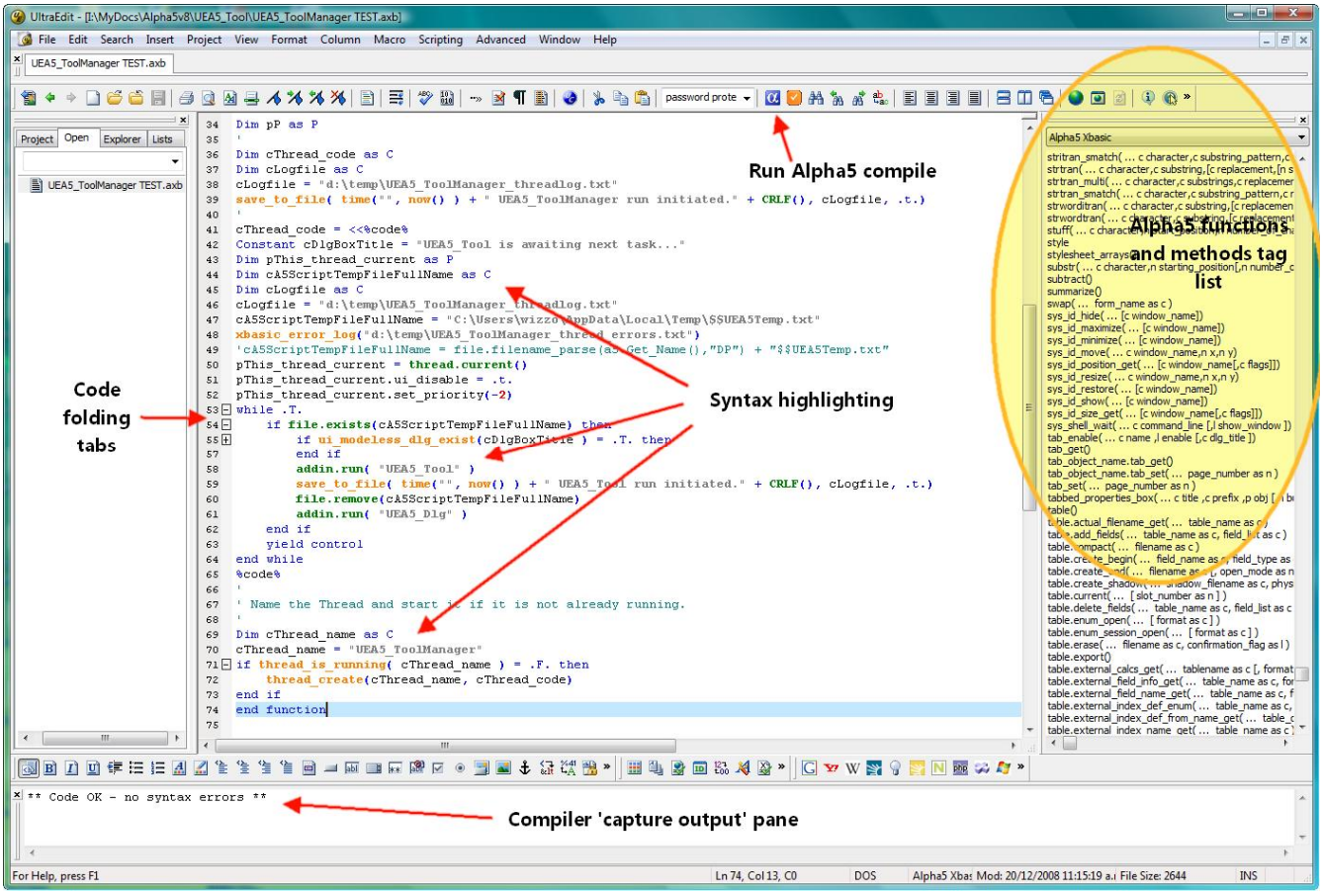


Figure 10: *PGPre* implementation in UltraEdit with Syntax Highlighting & Function Tag List and more…

Figure 10 shows:

- The text editor panel (large one in the centre) where you compose and modify your Alpha5 Xbasic code.
- The “Tags” list on the right where all Alpha5 functions and Methods are shown and can be copied into your current Xbasic script by double click on your selection.
- “Run Alpha5 compile” shows where a toolbar icon can be clicked to run an Alpha5 Xbasic compile.
- Code folding expansion and contraction tabs and syntax highlighting.
- Compiler “capture output” pane where the Alpha5 compiler output is displayed.

Additional Features - Syntax Highlighting and Function Tag List

Text editing applications for software developers typically provide syntax highlighting, code folding, indentation and function tag list related features. These features are product-dependant and language-dependant.

Support for these features is currently provided for UltraEdit/UEStudio by template files available from the Alpha5 Code Archive Forum. Support for Notepad++ is provided in the ***PGPre*** installation folder, for example “C:\Program Files\PGSD\PGPreCompiler \Config\NotePad++” that is created when the PGPre setup installer is run. There is a ReadMe file located there that provides further information.

Web Site Links

The following web sites are referenced in this document.

Alpha Software	http://www.alphasoftware.com
Alpha Software Users’ forum	http://msgboard.alphasoftware.com/alphaforum/
Alpha5 Code Archive Forum	http://msgboard.alphasoftware.com/alphaforum/forumdisplay.php?f=24
Notepad++	http://notepad-plus.sourceforge.net/uk/site.htm
WinMerge	http://winmerge.org
Beyond Compare	http://www.scootersoftware.com
UltraEdit and UEStudio	http://www.ultraedit.com
Subversion	http://subversion.tigris.org/
TortoiseSVN	http://tortoisesvn.tigris.org/
QVCS	http://www.qumasoft.com

Support and Suggestions

For support and improvement suggestions, contact the developer.
Brett Sinclair
PGSD Software
mailto: info@pgsd.co.nz
http://www.pgsd.co.nz

Dated: 24 January 2011

Release Notes

Version 1.0 released 21 November 2008

Version 2.0 released 20 December 2008

- New re-organized and more compact menus.
- The PGPre menus can now be hidden completely.
- Directive defaults for closing Alpha5, minimize, run script and warn if script already exists can now be set from a Settings menu item.
- Compiler output can now be captured and displayed in an external editor's 'capture output' pane (requires compatible editor).
- Comprehensive new script Import and Export windows allow scripts to be displayed in a summary form and are automatically compared to the same named script text files in a nominated working folder.
- Separate detailed script-to-script-text file compares can be run directly from the Import and Export windows.
- Script Import and Export items added to main menu item.
- Compatibility features added for Subversion compatible file versioning systems.
- Many other tweaks and improvements.
- Intelligent processing of password protected scripts.

Version 2.1 released January 2009

- The PGPre setup installer now creates a Windows desktop shortcut to run PGPre and Alpha5.
- The "translate" option has been moved to the PGPre settings panel.
- PGPre actions that allow compare, import, export and deletion of scripts and script text files have been amalgamated into a single display panel. Formerly they were all separate with their own separate menu items.
- The PGPre main menu has been altered as a consequence of the previous change and a new item named "actions" added to the main and relocated menus.
- The new Actions display panel shows scripts on the left and matching script text files on the right and shows whether the two match or differ. The display can cycle between three display options: display scripts only, display text files only and display both.
- The compile feature has been upgraded and improved.
- Numerous other fixes and improvements.